Document made available under the Patent Cooperation Treaty (PCT)

International application number: PCT/AU05/000487

International filing date: 04 April 2005 (04.04.2005)

Document type: Certified copy of priority document

Document details: Country/Office: AU
Number: 2005901063
Filing date: 07 March 2005 (07.03.2005)

Date of receipt at the International Bureau: 26 April 2005 (26.04.2005)

Remark: Priority document submitted or transmitted to the International Bureau in compliance with Rule 17.1(a) or (b)

World Intellectual Property Organization (WIPO) - Geneva, Switzerland
Organisation Mondiale de la Propriété Intellectuelle (OMPI) - Genève, Suisse

Australian Government

Patent Office
Canberra

I, JANENE PEISKER, TEAM LEADER EXAMINATION SUPPORT AND SALES hereby certify that annexed is a true copy of the Provisional specification in connection with Application No. 2005901063 for a patent by GREGORY ALAN COLLA as filed on 07 March 2005.

WITNESS my hand this
Eighteenth day of April 2005

JANENE PEISKER
TEAM LEADER EXAMINATION
SUPPORT AND SALES

Method and Apparatus for the Operation of a Secure Email Directory Service

Inventors: Gregory Alan COLLA, Neville Robert JONES

Field of Invention

The invention relates to method and apparatus for the operation of a secure email directory service.

Background to the Invention

E-mail is an essential business tool for the operation modern business. It provides an easily used tool for the rapid transfer of information. Such information can be exchanged between people/machine and other people/machines.

A popular protocol for transferring e-mail is the Simple Message Transport Protocol (SMTP). This protocol is the recognised standard for the transport of messages across the Internet [RFC0821].

SMTP, due to its simple nature, is prone to a number of security threats. Amongst these is the threat to message confidentiality. When a message is transmitted between the sender and recipient, it passes through many intermediate devices such as email relays and network routers. These devices may belong to any number of third parties. Each of these parties has the opportunity to access the contents of the message. Neither the sender nor the recipient would have knowledge whether a third party had accessed, or copied the message.

Further threats to message security are to its authenticity, integrity and repudiability. Non-authentic messages can be easily generated whereby one entity can act as another so it appears as though the message was sent by the impersonated party.

The recipient cannot tell the origins of the message, just from the message itself. Furthermore, if a message is modified en-route between the sender and recipient, the recipient will not be able to easily identify the modification thereby putting in doubt the integrity of the whole message. Finally, the sender of a message can repudiate its transmission claiming that the were not the author of the message or that the contents had been altered, due to the weaknesses outlined.

To resolve these issues, extra security measures are required. Generally there are two solutions, secured networks or secured messages. In the first, messages can be exchanged on secure networks, whereby confidentiality, authenticity, integrity and non-repudiation are assured to some degree by characteristics of the network layout, protection mechanisms or communications protocol. No specific security additions are required for the email systems on these networks, as the environment is regarded as secure. In the latter, the message itself can be secured and safely exchanged over an insecure network. This requires additional security features in the email agent, so that it can generate and interpret secure messages.

In general, most organisation's networks operate using the secured network philosophy. Messages between staff in a single organisation do not need extra security beyond that provided by the isolated nature of their private network.

Messages that traverse the Internet and that require additional security typically use the secured message approach. This is because it is typically difficult to establish secure networks between perimeter email relays of sender and recipient on an ad hoc basis; instead the sender and recipient agree on a way the message can be sent in a secured format and the intermediate email systems simply deliver it as if it were a normal email message.

Email clients capable of sending secured messages are often capable of querying Internet directory services to automatically discover the means to secure messages for a particular recipient.

With the rise in the amount of malicious content being transmitted via email, many organisations are choosing to use technology at their perimeter email gateways to scan inbound (and sometimes outbound) email messages to ensure that items such as viruses, spam and pornographic material do not enter the network. These email gateways are becoming increasingly sophisticated, but secured messages present specific challenges to them. The gateway, if it is to scan encrypted messages for prohibited content, needs to have knowledge of the secure message format being used and have the key to decrypt the message.

Discussion of Prior Art

The prior art discussed herein describe the various approaches of using public key cryptography to secure messages, at the data level, between sender and recipient when transmitted across insecure networks.

PGP

Pretty Good Privacy (PGP) [RFC1991] [RFC2440] was first developed in the early 1990s. It is a form of package security whereby an arbitrary blob of data can be secured using the application of public key cryptographic techniques. When integrated with email software, PGP can simply take the content of an email message as its input blob of data and secure it for the recipient of the message, using the recipient's public key to uniquely encrypt it for them.

It has since become very popular with the Internet community because of its low price, ease of use and its convenient 'web-of-trust' model for verification of public key associations with individuals. In this model, any member of the community can indicate whether a public key belongs to an individual and hence if it is trustworthy. If a person that the sender trusts has indicated that a public key belongs to the recipient , then the sender will trust the recipient's public key. Such a model works well for small communities but there are scalability issues that limits its uptake in large communities. PGP users can search for recipient public keys in PGP Key Servers on the Internet.

Secure email gateways using PGP technology have been developed.

PEM

Privacy Enhanced Mail (PEM) [RFC1421] was proposed in the early-1990s and like PGP uses public key cryptography to secure an email message. It differs from PGP in that it uses a trusted third party security model to improve scalability. In this model a trusted third party issues the recipient's public key in the form of a digital certificate. Now the sender does not have to use a 'web-of-trust' to verify the recipient's public key before sending – they can simply verify that the trusted third party issued it.

S/MIME

Secure Multipart Internet Mail Extensions (S/MIME) [RFC2311] was developed in the late 1990s and was the integration of a public key cryptosystem using trusted third parties with the MIME standard. S/MIME is based on RSA's PKCS#7 data standard [PKCS#7].

S/MIME uses X.509 certificates [X.509] for electronic authentication of remote entities. The certificate contains a public key that is used to verify digital signatures and/or encrypt messages. The certificate also contains the entity's name and email address [RFC2312].

Generally, certificates may be exchanged between parties in two ways. In the first mechanism, one party sends a signed message to the second. The signed message includes the sender's certificate(s) and the receiving party extracts it from the message and stores in an electronic email address book. The second party can then use the public encryption key from the first party's certificate to generate an encrypted message for the first party. The second mechanism involves the publication of certificates in an electronic directory which is accessible to other entities on the network. The sending party retrieves the intended recipient's certificate using a directory access protocol, e.g. Lightweight Directory Access Protocol (LDAP) [RFC2251] or using a web application connected with Hypertext Transfer Protocol [RFC2616]. Depending on the configuration of the directory, a remote user may be able to browse for certificates, or may be able to search for certificates using the recipient's email address, name, organisation, etc., as a search parameter. The certificate search and retrieval mechanism is often a feature of secure email clients.

S/MIME has been integrated into email clients such as Microsoft Outlook, Lotus Notes. It has also been implemented as a "plug-in" by third party developers, for example by Baltimore Technologies.

Email Gateways and S/MIME

S/MIME has been integrated into secure email gateways. Email gateways join email networks. A secure email gateway can be used to join a secure and an insecure network. On the secure network, typically the organisation's network, the message travels "in-the-clear", as it does not require further security. On the insecure network, the message requires additional security, such as that available from S/MIME.

The secure email gateway can act as a security proxy for entities on the secure network. The secure email gateway should be able to sign and/or encrypt messages crossing from the secure network to insecure network, and decrypt and/or verify messages crossing from the insecure network to the secure network. Secure email gateways remove the need for entities on secure networks to use and manage their own keys and encrypted email which reduces costs for the organisation and relieves the end-user of much of the complexity of managing secure email.

Nonetheless, there are problems associated with the operation of secure email gateways. At the forefront of these is the difficulty a sender has in structuring a secure email message that is to be delivered to a recipient in an organisation but which is to be decrypted at the organisation's secure email gateway. When using certificate directory services it is common for the email client to retrieve a recipient's encryption certificate by using the recipient's email address as the search parameter. In the case of secure email gateway operation, the sender may not be able to retrieve the email gateway's encryption certificate, as there is no standard mechanism to retrieve this from a directory, based on the recipient's email address.

The S/MIME Certificate Handling standard [RFC2312] also causes interoperability problems between S/MIME clients and secure email gateways. Specifically, when a secure email client verifies a message that is signed by a secure email gateway on behalf of another entity, it should warn the user that the signer's email address does not match the sender's email address.

Domain Security Services using S/MIME [RFC3183] defines a certificate profile for secure email gateways, gateway behaviour, and how secure email clients interoperate with these gateways. This does not resolve the two problems previously described, as few existing secure email clients implement the protocol.

Version 3.1 of the S/MIME Certificate Handling protocol [RFC3850] specifies that certificates that do not contain email addresses have no requirements for verifying the email address associated with the certificate. This does not resolve the problem of backwards compatibility with existing S/MIME clients, it degrades the security of message verification, and does not resolve the issue of encryption certificate retrieval.

It is an object of the present invention to ameliorate problems with prior art solutions.

Table of Contents

Table of Figures

References

3DES FIPS PUB 46-3: DATA ENCRYPTION STANDARD (DES), NIST, 1999, http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf

BCCrypto http://www.bouncycastle.org

certstore-http Gutmann, Internet X.509 Public Key Infrastructure Operational Protocols: Certificate Store Access via HTTP, IETF, 2005, http://www.ietf.org/internet-drafts/draft-ietf-pkix-certstore-http-08.txt

gentoo http://www.gentoo.org

mySQL http://www.mysql.org

openldap http://www.openldap.org

openssl http://www.openssl.org

PKCS#7 http://www.rsasecurity.com/rsalabs/pkcs/pkcs-7/index.html

RFC0821 Postel, Simple Mail transport Protocol, IETF, 1982, http://www.ietf.org/rfc/rfc0821.txt

RFC1421 Linn, Privacy enhancement for Internet electronic mail: Part I: Message encipherment and authentication procedures, IETF, 1993, http://www.ietf.org/rfc/rfc1421.txt

RFC1991 Atkins, et al, PGP Message Exchange Formats, IETF, 1996, http://www.ietf.org/rfc/rfc1991.txt

RFC2251 Wahl, et al, Lightweight Directory Access Protocol (v3), IETF, 1997, http://www.ietf.org/rfc/rfc2251.txt

RFC2311 Dusse, et al, S/MIME Version 2 Message Specification, IETF, 1998, http://www.ietf.org/rfc/rfc2311.txt

RFC2312 Dusse, et al, S/MIME Version 2 Certificate Handling, IETF, 1998, http://www.ietf.org/rfc/rfc2312.txt

RFC2440 Callas, et al, OpenPGP Message Format, IETF, 1998, http://www.ietf.org/rfc/rfc2440.txt

RFC2510 Adams, et al, Internet X.509 Public Key Infrastructure - Certificate Management Protocols, IETF, 1999, http://www.ietf.org/rfc/rfc2510.txt

RFC2616 Fielding, et al, Hypertext Transfer Protocol – HTTP/1.1, IETF, 1999, http://www.ietf.org/rfc/rfc2616.txt

RFC2633 Ramsdell (ed), S/MIME Version 3 Message Specification, IETF, 1999, http://www.ietf.org/rfc/rfc2633.txt

RFC3183 Dean, et al, Domain Security Services using S/MIME, IETF, 2001, http://www.ietf.org/rfc/rfc3183.txt

RFC3280 Housley, et al, Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, IETF, 2002, http://www.ietf.org/rfc/rfc3280.txt

RFC3850 Ramsdell (ed), Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.1 Certificate Handling, IETF, 2004, http://www.ietf.org/rfc/rfc3850.txt

SHA-1 FIPS PUB 180-1: Secure Hash Standard, NIST, 1995, http://csrc.nist.gov/publications/fips/fips180-2/fips180-1.pdf

smime-certcapa Santesson, Certificate extension for S/MIME Capabilities, IETF, 2005, http://www.ietf.org/internet-drafts/draft-ietf-smime-certcapa-03.txt

X.509 The Directory: Public-key and attribute certificate frameworks, ITU, 2000, http://www.itu.int/home/index.html

Summary of the Invention

In accordance with an embodiment of the invention there is provided method and apparatus for the provision of digital identifiers.

In accordance with a further embodiment of the invention said digital identifiers are provided by an Identification Provider.

In accordance with a further embodiment of the invention such digital identifiers are public key containers.

In accordance with a further embodiment of the invention such public key containers may be digital certificates [X.509].

In accordance with a further embodiment of the invention such public key containers may be PGP public keys [RFC2440].

In accordance with a further embodiment of the invention said Identification Provider accepts requests for said digital identifiers using a directory access protocol.

In accordance with a further embodiment of the invention said directory access protocol is the Lightweight Directory Access Protocol (LDAP) [RFC2251].

In accordance with a further embodiment of the invention said directory is accessible using Hypertext Transfer Protocol (HTTP) [RFC2616] [certstore-http].

In accordance with a further embodiment of the invention said directory is accessible using the Simple Mail Transfer Protocol (SMTP) [RFC0821].

In accordance with a further embodiment of the invention said Identification Provider provides public key containers that embed a domain's email gateway public key.

In accordance with a further embodiment of the invention said Identification Provider provides public key containers that embed an email address.

In accordance with a further embodiment of the invention said Identification Provider provides public key containers that embed an email sender's email address.

In accordance with a further embodiment of the invention said Identification Provider provides public key containers that embed an email recipient's email address.

In accordance with a further embodiment of the invention said Identification Provider embeds any requested email address in said provided public key containers.

In accordance with a further embodiment of the invention said Identification Provider generates, on an as needed basis, domain proxy public key containers.

In accordance with a further embodiment of the invention said domain proxy public key containers embed a public key belonging to the domain's email gateway and an email address other than that of the domain's email gateway.

In accordance with a further embodiment of the invention said Identification Provider generates, on an as needed basis, domain inbound encryption proxy public key containers.

In accordance with a further embodiment of the invention said domain inbound encryption proxy public key containers embed a domain's email gateway public key, an email address of a member of the domain and may embed a parameter that indicates their use is restricted to encryption functions.

In accordance with a further embodiment of the invention said Identification Provider generates, on an as needed basis, domain outbound encryption proxy public key containers.

In accordance with a further embodiment of the invention said domain outbound encryption proxy public key containers embed a domain's email gateway public key, an email address of a member of another domain and may embed a parameter that indicates their use is restricted to encryption functions.

In accordance with a further embodiment of the invention said Identification Provider generates, on an as needed basis, domain signing proxy public key containers.

In accordance with a further embodiment of the invention said domain signing proxy public key containers embed public key belonging to a domain's email gateway, an email address of a member of the domain and may embed a parameter that indicates their use is restricted to signing functions.

In accordance with a further embodiment of the invention said email gateway public key can be used by an external sending party for inbound message encryption, whereby the sender requests a public key container using the recipient's email address and the Identification Provider returns a domain inbound encryption proxy public key container embedding the gateway's public key and the recipient's email address.

In accordance with a further embodiment of the invention said email gateway public key can be used by a internal sending party for outbound message encryption, whereby the sender requests a public key container using the recipient's email address and the Identification Provider returns a domain outbound encryption proxy public key container embedding the gateway's public key and the recipient's email address.

In accordance with a further embodiment of the invention the Identification Provider generates the domain proxy public key container embedding an email gateway public key and recipient email address using knowledge of the email gateway's public key and the recipient's email address as derived from the request parameter(s).

In accordance with a further embodiment of the invention the Identification Provider determines whether to generate a domain inbound encryption, outbound encryption or signing proxy public key container from the nature of, or additional parameters in the request.

In accordance with a further embodiment of the invention the requester of domain proxy public key containers must authenticate to the Identification Provider to obtain public key containers from it.

In accordance with a further embodiment of the invention a requester of domain proxy public key containers that is determined by the Identification Provider to be external to the domain due to its authentication credentials shall have domain inbound encryption proxy public key containers returned to it.

In accordance with a further embodiment of the invention a requester of domain proxy public key containers that is determined by the Identification Provider to be a member of the domain due to its authentication credentials shall have domain outbound encryption proxy public key containers returned to it.

In accordance with a further embodiment of the invention a requester of domain proxy public key containers that is determined by the Identification Provider to be the domain email gateway due to its authentication credentials shall have domain signing proxy public key containers returned to it.

In accordance with a further embodiment of invention said domain proxy public key container shall be traceable to the Identification Provider.

In accordance with a further embodiment of the invention said domain encryption proxy digital certificates (inbound or outbound) may contain the same Serial Number and Issuer name as the email gateway's digital certificate so as to provide backwards compatible operation for those gateways which use these parameters to determine which private key to use to decrypt the message.

In accordance with a further embodiment of the invention said email gateway requests domain signing proxy public key containers from said Identification Provider.

In accordance with a further embodiment of the invention said email gateway requests domain signing proxy public key containers from said Identification Provider using LDAP or the Certificate Management Protocol (CMP) [RFC2510].

In accordance with a further embodiment of the invention said email gateway digitally signs outbound message on behalf of a sender using said email gateway's private key and attaches said domain signing proxy public key container with sender's email address, whereby said public key embedded in said public key container can be used for message verification by the recipient.

In accordance with a further embodiment of the invention said Identification Provider provides domain proxy public key containers that embed the security preferences of the email gateway.

In accordance with a further embodiment of the invention said security preferences may be S/MIME Capabilities [smime-certcapa].

In accordance with a further embodiment of the invention said security preferences may be OpenPGP Symmetric Algorithm Preferences [RFC2440].

Detailed Description of the Invention

Description of System Deployment

The Secure Messaging System consists of the following components (refer Figure 1: System Deployment for Secure Email Gateway and Domain Directory Service).

(1) An insecure network. Messages on this network require further levels of security such as digital signatures and encryption, to be treated as authentic and confidential.

(2) A secure network. Cleartext messages on this network require no further security. Threats to message confidentiality and authenticity within this network are treated as low risk.

(3) A Secure Email Gateway that bridges the insecure (1) and secure (2) networks. The Secure Email Gateway has a public/private key pair used for message signing, and a public/private key pair for message encryption. The same key pair may be used for signing and encryption. The Secure Email Gateway is capable of performing directory requests against the Secure Email Domain Directory Service (6).

(4) A secure email client on the insecure network (1). The secure email client has the capability to encrypt and decrypt messages, to sign and verify messages and perform directory requests for public key containers in the Secure Email Domain Directory Service (6).

(5) An email client on the secure network (2) which is not necessarily a secure email client.

(6) A "Secure Email Domain Directory Service" (SEDDS), which is trusted by the secure email client (4), and produces public key containers on

behalf of entities on the secure network (5). The SEDDS has access to the Email Gateway's public signing key and public encryption key.

An alternative deployment may replace the secure email client (4) with a further Secure Email Gateway that bridges the insecure network (1) with a further secure network (not shown).

A further alternative deployment may add further directory services (not shown) on the insecure network (1) that the Secure Email Domain Directory Service (6) can access.

Intrinsic to the email network are email servers, email relays and domain name servers. These are used to transport the message from the sender to the recipient. The invention is described without these components.

Description of System Operation

Description of Message Decryption by Secure Email Gateway

Message Inbound to Secure Network

A user on the insecure network (the sender) wishes to send a message to a user on the secure network (the recipient). The sender wishes to ensure the contents of the message remains confidential, at least while in transit over the insecure network.

The sender generates the message using the secure email client on the insecure network (Figure 1-4). The sender specifies the recipient's email address and specifies that the message is to be encrypted.

Prior to transmission, the secure email client (Figure 1-4) performs an authenticated request (an anonymous request is regarded as a special case of authenticated request) for the public key container(s) of the recipient against the Secure Email Domain Directory Service (Figure 1-6). The Secure Email Domain Directory Service determines that the request has originated from a client that is external to the recipient domain and consequently generates a domain inbound encryption proxy public key container in response. The domain inbound encryption proxy public key container embeds the Secure Email Gateway's (Figure 1-3) public encryption key and the recipient's email address. This public key container is returned to the secure email client which in turn, uses the public key from the container to encrypt the message.

The secure email client transmits the encrypted message to the recipient.

The mail system on the insecure network (Figure 1-1) routes the message to the Secure Email Gateway.

The Secure Email Gateway (Figure 1-3) decrypts the message using its private decryption key. It forwards the decrypted message to the recipient.

The mail system on the secure network (Figure 1-2) routes the message to the recipient's mailbox. The recipient retrieves the message using the mail client on the secure network (Figure 1-5) and the recipient reads the message.

Message Outbound from Secure Network

A user on the secure network (the sender) wishes to send a message to a user on the insecure network (the recipient). The sender wishes to ensure the contents of the message remains confidential, even while in transit over the secure network.

The sender generates the message using the secure email client on the secure network (Figure 1-4). The sender specifies the recipient's email address and specifies that the message is to be encrypted.

Prior to transmission, the secure email client (Figure 1-4) performs a simple username and password authenticated request for the public key container(s) of the recipient against the Secure Email Domain Directory Service (Figure 1-6). The Secure Email Domain Directory Service determines that the request has originated from a client within the secure network and consequently generates a domain outbound encryption proxy public key container in response. The domain outbound encryption proxy public key container embeds the Secure Email Gateway's (Figure 1-3) public encryption key and the recipient's email address. This public key container is returned to the secure email client which in turn, uses the public key from the container to encrypt the message.

The secure email client transmits the encrypted message to the recipient.

The mail system on the secure network (Figure 1-1) routes the message to the Secure Email Gateway.

The Secure Email Gateway (Figure 1-3) decrypts the message using its private decryption key and performs any content analysis tasks required. It can then re-encrypt the message for the recipient using the recipient's genuine public key or it may even forward the decrypted message to the recipient.

The mail system on the insecure network (Figure 1-2) routes the message to the recipient's mailbox. The recipient retrieves the message using the mail client on the insecure network (Figure 1-5) and the recipient reads the message.

Description of Message Signing by Secure Email Gateway

A user on the secure network (the sender) wishes to send a message to a user on the insecure network (the recipient). The sender wishes that the recipient be able to verify the message has originated from the domain and to be assured that it has retained its integrity while in transit over the insecure network.

The sender generates the message using the email client on the secure network (Figure 1-5). The sender specifies the recipient's email address and sends the message. The mail system on the secure network (Figure 1-2) routes the message to the Secure Email Gateway (Figure 1-3).

The Secure Email Gateway signs the message with its private signing key. It performs an authenticated request for the public key container(s) of the sender against the Secure Email Domain Directory Service (Figure 1-6). The Secure Email Domain Directory Service determines that the request has originated from the domain's Secure Email Gateway and in consequence generates a domain signing proxy public key container in response. The domain signing proxy public key container has the Secure Email Gateway's public signing key and the sender's email address, and a parameter that indicates the container is for use with digital signature operations. The Secure Email Gateway attaches the signing public key container to the message. The Secure Email Gateway relays the message to the recipient's email mailbox across the insecure network (Figure 1-1).

The recipient retrieves the message from the email mailbox using the secure email client on the insecure network (Figure 1-4). The secure email client verifies the authenticity of the message.

Description of Components

Description of Operation of the Secure Email Domain Directory Service

The Secure Email Domain Directory Service consists of the following sub-components (refer Figure 2: Sequence diagram depicting operation of Domain Directory Service):

(2) A directory access interface;

(3) A controller;

(4) A certification authority, with private signing key; and

(5) A database or data store, which stores gateway public keys and caches generated public key containers.

A requester (Figure 2-1) initiates the provision of public key containers. The requester could be a secure email gateway or a secure email client.

The Secure Email Domain Directory Service operates as follows (refer Figure 2: Sequence diagram depicting operation of Domain Directory Service). For simplicity the provision of a single public key container is described, however generation of multiple containers for various purposes is quite possible.

1. Request Public Key Container - (Figure 2-6)

   The requesting client (Figure 2-1) requests a public key container(s) from the Secure Email Domain Directory Service via the directory access interface (Figure 2-2). It first binds to the directory using a simple username and password combination, or may perform an anonymous bind where it does not supply any authentication credentials. It then specifies the entity for which it requires public key container(s) by the entity's email address. The search request contains information about what type of public key container is required by the client (either X509 certificate or PGP public key) in terms of the directory attribute that is to be returned by the server.

2. Notify Controller - (Figure 2-7)

   The directory access interface (Figure 2-2) notifies the controller (Figure 2-3) of the request, including the entity's email address, the authentication credentials of the requesting client and the type of public key container that is required. The controller (Figure 2-3) determines the origin of the request by comparing the presented authentication credentials with known, allocated sets of credentials. Anonymous requests are deemed to originate from clients on the insecure network (Figure 1-4) while authenticated requests may either originate from clients within the secure network (Figure 1-5) or from the gateway itself (Figure 1-3). To distinguish the two, the clients within the secure network use different credentials to those used by the email gateway. The controller knows the destination domain based on the email address in the search and hence can determine the email gateway(s) that will be traversed by an email message going from origin to destination. From this knowledge it can decide which form of proxy public key container to construct, either inbound encryption proxy, outbound encryption proxy or signing proxy.

3. Retrieve Gateway Public Key - (Figure 2-8)

If the controller has determined that it is to return a domain inbound encryption proxy public key container then it uses the domain information from the email address in the client search request as a parameter to retrieve the domain gateway's public key from the trusted database (Figure 2-5). If the controller has determined that it is to return a domain outbound encryption proxy public key container or a domain signing proxy public key container then it uses the domain information associated with the request's origin (either domain member or domain gateway) as a parameter to retrieve the domain gateway's public key from the trusted database (Figure 2-5). Alternatively, the controller retrieves the email gateway's public key container from another directory or database, verifies the authenticity of the gateway's public key container and extracts the public key.

4. Generate Public Key Container (Figure 2-9)

The controller (Figure 2-3) generates a public key container using the certification authority (Figure 2-4). It specifies the public key as retrieved, the email address of the requested entity and the type of public key container to create. It may optionally add details of the email security preferences of the gateway to the container.

5. Return Public Key Container to Controller (Figure 2-10)

   The certification authority (Figure 2-4) returns the generated public key container to the controller (Figure 2-3).

6. Cache generated public key container ((Figure 2-11)

   The controller (Figure 2-3) optionally stores the generated public key container in the database (Figure 2-5).

7. Return Public Key Container to Directory Access Interface (Figure 2-12)

   The controller (Figure 2-3) returns the generated public key container to the directory access interface (Figure 2-2).

8. Return Public Key Container to Requester (Figure 2-13)

   The directory access interface (Figure 2-2) returns the generated public key container to the requesting client (Figure 2-1).

Preferred Implementation

FORMATS, INTERFACES AND PROTOCOLS

Request between Secure Email Gateway and Secure Email Domain Directory Service uses LDAP v3 [RFC2251] protocol.

Message transmission protocol is SMTP [RFC0821].

Message security is provided with S/MIME v3 [RFC2633].

Public key containers are X.509 v3 [X.509] certificates.

Message digest is SHA-1 [SHA-1].

Message encryption is 3DES [3DES].

Email security preferences returned in the X.509 certificates consist of the appropriate standard Key Usage (KeyUsage = digitalSignature and/or keyEncipherment or keyAgreement [RFC3280]) and Extended Key Usage (KeyPurposeId = id-kp-emailProtection [RFC3280])extensions, and optionally the S/MIME Capabilities private extension [smime-certcapa].

COMPONENTS

Email client is Microsoft Outlook Express 6, operating on a Microsoft Windows XP platform.

Secure Email client is Microsoft Outlook Express 6, operating on a Microsoft Windows XP platform. It uses a 1024 bit RSA private/public key pair. The public key is embedded in an X.509 certificate generated by OpenSSL [openssl].

Secure Email Gateway is Clearswift MAILsweeper 5.0 integrated with SSS SecureIT S/MIME component. The Secure Email gateway operates on Microsoft Windows 2000 Server on an Intel Pentium 4 platform. It uses a 1024 bit RSA private/public key pair. The public key is embedded in an X.509 certificate generated by OpenSSL.

The Secure Email Domain Directory Service is built using the Bouncy Castle Crypto package [BCCrypto] on the Java Standard Development Kit 1.4.2 for the Certification Authority, openLDAP 2.2.8 [openldap] for the directory access interface and mySQL 4.0.18 [mySQL] for the database. It operates using the Gentoo Linux 2004.3 [gentoo] operating system on an Intel Pentium 4 platform.

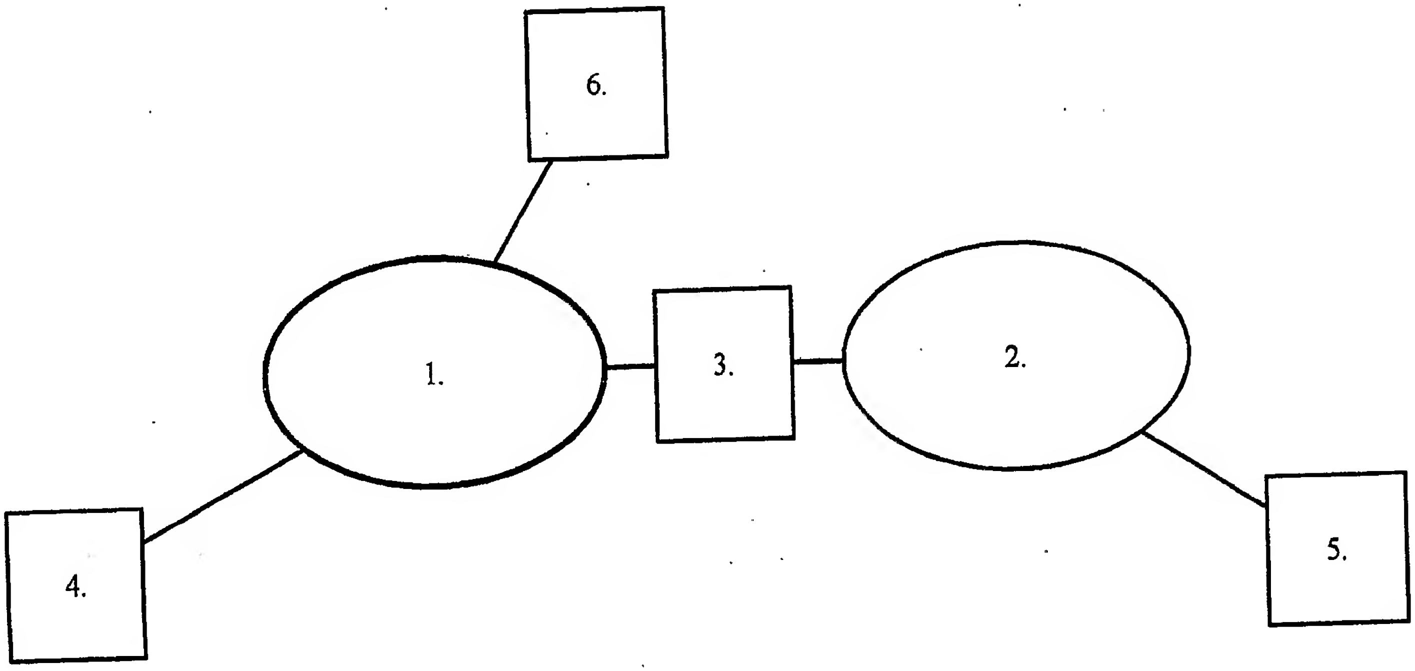

Figure 1: System Deployment for Secure Email Gateway and Domain Directory Service

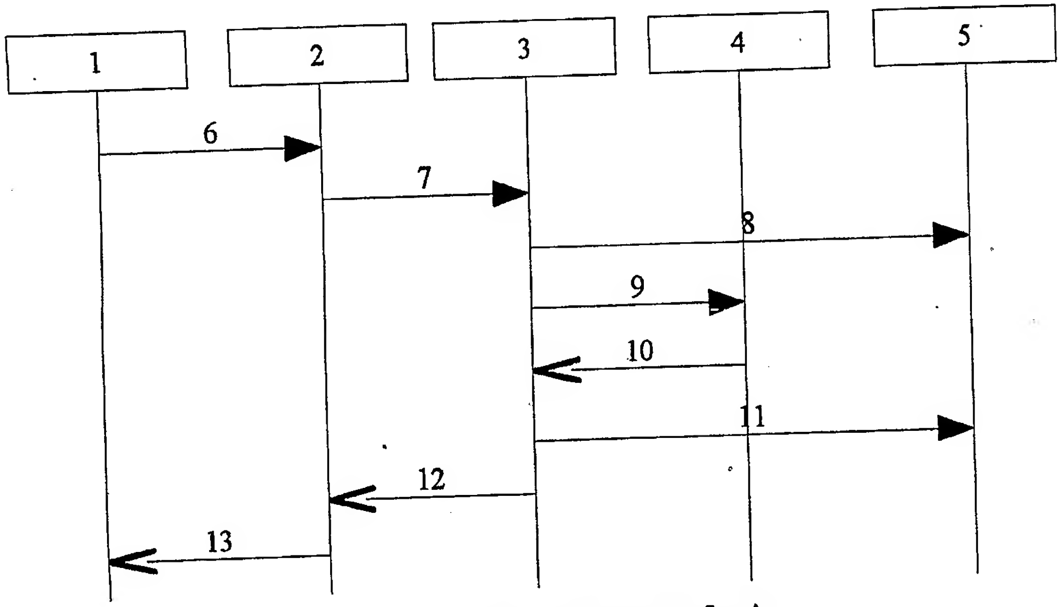

Figure 2: Sequence diagram depicting operation of Domain Directory Service